2772/AF

Patent

Docket No.: SGI 15-4-453.00

#15 Appeal Brief 1-23-01

I hereby certify that this transmittal of the below described document is being deposited with the United States Postal Service in an envelope bearing First Class Postage and addressed to the Commissioner of Patents and Trademarks, Washington, D.C., 20231, on the below date of deposit.					
Date of Deposit:	01/08/00	Name of Person Making the Deposit:	KATHERINE RINALDI	Signature of the Person Making the Deposit:	Katherine Rinaldi

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of: Matthew Papikipos, Phil Gossett, Christian Pappas, Henry Moreton and Robert Williamson

Serial No.: 08/845,526

Examiner: Padmanabham, M.

Filed: 04/25/97

Art Unit: ~~2772~~ 2671

For: A METHOD AND SYSTEM FOR EFFICIENTLY DRAWING NURBS SURFACES FOR 3D GRAPHICS

The Commissioner of Patents and Trademarks
Washington, D.C. 20231
Sir:

Transmittal of an Appeal Brief
(Under 37 CFR §1.192)

☒ Transmitted herewith, in triplicate, is the APPEAL BRIEF in this application with respect to the Notice of Appeal filed on: 11/07/00

X The application is on behalf of other than a small entity
.... The application is on behalf of a small entity.
.... A verified statement of small entity status is attached.
.... A verified statement of small entity status has been previously filed herein.

Fee Calculation (for other than a small entity)		
Filing Appeal Brief	$280	$280.00
Total Fees		$280.00

PAYMENT OF FEES

1. The full fee due in connection with this communication is _ provided as follows:

[X] The Commissioner is hereby authorized to charge any additional fees associated with this communication or credit any overpayment to Deposit Account No.: 23-0085.
A duplicate copy of this authorization is enclosed.

[ ] A check in the amount of $

[ ] Charge any fees required or credit any overpayments associated with this filing to Deposit Account No.: 23-0085.

RECEIVED
JAN 1 9 2001
Technology Center 2600

Please direct all correspondence concerning the above-identified application to the following address:

WAGNER, MURABITO & HAO LLP
Two North Market Street, Third Floor
San Jose, California 95113
(408) 938-9060

Respectfully submitted,

Date: 01/08/01

By: [signature]
Glenn D. Barnes
Reg. No. 42,293

OIPE JC141
JAN 16 2001
PATENT & TRADEMARK OFFICE

SGI 15-4-453 PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant : Papakipos et al.

Serial No. : 08/845,526 Group Art Unit : ~~2772~~ 2671

Filed : 04/25/97 Examiner : Padmanabhan, M.

For : A METHOD AND SYSTEM FOR EFFICIENTLY DRAWING NURBS SURFACES FOR 3D GRAPHICS

BRIEF ON APPEAL

Assistant Commissioner for Patents & Trademarks
Washington, D.C. 20231

Sir:

REAL PARTY IN INTEREST

The real party in interest is SGI, Inc. (formerly known as Silicon Graphics, Inc.)

RECEIVED
JAN 19 2001
Technology Center 2600

STATUS OF CLAIMS

Claims 1, 2, 6-9, 12-16, and 18-25, which stand under final rejection, are currently pending and are the subject of this appeal. No other claims are pending.

STATUS OF AMENDMENTS

An Amendment was filed on June 19, 2000 in response to the Office Action mailed on January 18, 2000. The Amendment was entered. There has been no amendment after the Final Rejection mailed on August 1, 2000.

00000078 230085 08845526
310.00 CH

SUMMARY OF INVENTION

The present invention, as embodied in Claims 1, 2, 6-9, 12-16, and 18-25, is directed towards a computer implemented method for rendering a NURBS defined curve or surface without first converting the NURBS defined curve or surface to a polygon mesh. The method is embodied within a computer system having a processor, a bus, and a graphics rendering pipeline for displaying 3D graphics on a display (e.g., see figure 15).

The rendering of the curved lines and surfaces is viewed as 3D graphics on a display of the computer system (e.g., a CAD workstation). The curves and surfaces are parametric, wherein they are modeled by non-uniform rational B-splines (NURBS). The process of the present invention functions by receiving a NURBS model for rendering from a software program running on the one or more host processors (e.g., processors 1502 of figure 15). The NURBS model defines a curve or surface. The process efficiently converts the NURBS model to a Bezier model using the hardware of the graphics rendering pipeline (e.g., graphics rendering pipeline 301 of figure 15). The Bezier model describes the same curve or surface. The process of the present invention subsequently generates a plurality of points on the curve or surface using the Bezier model and the graphics rendering pipeline. The points on the surface are then used by the graphics rendering pipeline to render the curve or surface defined by the Bezier model. In so doing, the present invention provides a fast and efficient process for rendering NURBS models, hence, greatly increasing the speed of the graphics rendering process.

Independent Claim 1 recites a computer implemented method for rendering a NURBS defined curve or surface without first converting the NURBS defined curve or surface to a polygon mesh. The method includes the step of receiving a NURBS model for rendering from a software program running on the processor (e.g., processor 1502) of the computer system (e.g., computer system 1500). The NURBS model is a parametric representation of a curved surface. The NURBS model is converted to a Bezier model using the graphics rendering pipeline (e.g., graphics rendering pipeline 301). A plurality of Bezier control points are generated from a corresponding plurality of NURBS control points of the NURBS model using a tri-linear interpolator (e.g., trilinear interpolator 500 shown in figure 5) in the graphics pipeline (e.g., within texture mapping logic 309 shown in figure 4, which comprises a portion of the raster subsystem 304 shown in figure 15). The plurality of Bezier control points are generated from the corresponding plurality of NURBS control points of the NURBS model by using the plurality of NURBS control points as inputs to the tri-linear interpolator, and evaluating the NURBS control points to obtain each of the plurality of Bezier control points. Additionally, a plurality of points on a curve or surface are generated, wherein the curve or surface is defined by the Bezier model, using the graphics rendering pipeline. The curve or surface defined by the NURBS model is rendered using the plurality of points and using the graphics rendering pipeline such that the curve or surface is rendered without first converting the NURBS model to a polygon mesh.

Independent Claim 9 recites a computer implemented method for rendering a NURBS defined curve or surface without first converting the NURBS defined curve or surface to a polygon mesh. As with independent Claim 1, the method of Claim 9 is implemented using a graphics rendering pipeline of a computer system (e.g., computer system 1500). Claim 9 recites the step of implementing a de Casteljau process in the graphics pipeline (e.g., graphics rendering pipeline 301). A Bezier curve or surface is evaluated using the de Casteljau process implemented using a tri-linear interpolator (e.g., trilinear interpolator 500) included in the graphics pipeline by loading inputs of the tri-linear interpolator with a plurality of control points of the Bezier curve or surface, and generating a plurality of points on the curve or surface using the tri-linear interpolator. The Bezier curve or surface is rendered without first converting the Bezier curve or surface to a polygon mesh.

Independent Claim 13 recites a computer implemented method for converting a NURBS (non-uniform rational B-spline) curve or surface to a Bezier curve or surface using the graphics rendering pipeline. As with independent Claims 1 and 9, the method of Claim 13 is implemented using a graphics rendering pipeline of a computer system(e.g., computer system 1500). Claim 13 recites the step of loading a plurality of NURBS control points of a NURBS curve or surface into the graphics rendering pipeline (e.g., graphics rendering pipeline 301). The plurality of control points are evaluated into a resulting plurality of Bezier control points using a tri-linear interpolator included in the graphics rendering pipeline (e.g., trilinear interpolator 500). The Bezier curve or surface is generated using the resulting plurality of Bezier

control points. The Bezier curve or surface is rendered using a plurality of vertices derived from the plurality of Bezier control points.

Independent Claim 16 recites a computer implemented method for generating normal vectors (normals) for a surface As with independent Claims 1, 9, and 13, the method of Claim 16 is implemented using a graphics rendering pipeline of a computer system(e.g., computer system 1500). Claim 16 recites the step of generating a plurality of surface partials from the surface by loading inputs of a tri-linear interpolator (e.g., trilinear interpolator 500) included in a graphics rendering pipeline (e.g., graphics rendering pipeline 301) with a plurality of Bezier control points defining the surface. A plurality of surface tangents are generated from the plurality of surface partials using the graphics rendering pipeline. Additionally, at least one normal from the plurality of surface tangents is generated using the graphics rendering pipeline.

Independent Claim 20 recites a computer implemented method for using the graphics rendering pipeline to render a curve or surface directly from a NURBS (non-uniform rational B-spline) model. As with independent Claims 1, 9, 13, and 16, the method of Claim 20 is implemented using a graphics rendering pipeline of a computer system(e.g., computer system 1500). Claim 20 recites the step of performing a global to local transformation on a NURBS model using the graphics rendering pipeline. A plurality of NURBS control points are evaluated using tri-linear interpolation in the graphics rendering pipeline to obtain a plurality of points on a curve or surface defined by the NURBS model. The curve or surface is rendered using the plurality of points.

ISSUES

Issue 1: Whether Claims 1, 2, 6, 7, 8, 13, 14, and 15 are unpatentable under 35 U.S.C. § 103(a) over Jia et al., U.S. Patent No. 5,726,896 in view of Gharachorloo et al., U.S. Patent No. 5,488,684 in further view of Luken, Jr. U.S. Patent No. 5,278,948 in further view of Schulmeiss, U.S. Patent No. 5,717,847 in further view of Sherman et al., U.S. Patent No. 5,734,756.

Issue 2: Whether Claims 9 and 12 are unpatentable under 35 U.S.C. § 103(a) over Jia et al., U.S. Patent No. 5,726,896 in view of Luken, Jr. U.S. Patent No. 5,278,948 in further view of Schulmeiss, U.S. Patent No. 5,717,847 and in further view of Sherman et al., U.S. Patent No. 5,734,756.

Issue 3: Whether Claims 16, 18, and 19 are unpatentable under 35 U.S.C. § 103(a) over Jia et al., U.S. Patent No. 5,488,684 in view of Luken, Jr. U.S. Patent No. 5,278,948 in further view of Schulmeiss, U.S. Patent No. 5,717,847.

Issue 4: Whether Claims 20-25 are unpatentable under 35 U.S.C. § 103(a) over Luken, Jr. U.S. Patent No. 5,278,948 in view of Gharachorloo et al., U.S. Patent No. 5,488,684.

GROUPING OF CLAIMS

The rejected claims have been grouped together in each of the rejections. For each ground of rejection which Appellants contest herein and which applies

to more than one claim, such additional claims, to the extent separately identified and argued below, do not stand or fall together. Appellants have shown below the reasons why these claims are separately patentable.

ARGUMENTS

Issue 1: Whether Claims 1, 2, 6, 7, 8, 13, 14, and 15 are unpatentable under 35 U.S.C. § 103(a) over Jia et al., U.S. Patent No. 5,726,896 in view of Gharachorloo et al., U.S. Patent No. 5,488,684 in further view of Luken, Jr. U.S. Patent No. 5,278,948 in further view of Schulmeiss, U.S. Patent No. 5,717,847 in further view of Sherman et al., U.S. Patent No. 5,734,756.

Appellants respectfully contend that the claimed invention is not rendered obvious within the meaning of 35 U.S.C. Section 103(a) to a person having ordinary skill in the art to which the claimed invention pertains by the cited references.

Independent Claim 1

Independent Claim 1 recites a computer implemented method for rendering a NURBS defined curve or surface without first converting the NURBS defined curve or surface to a polygon mesh. The method includes the step of receiving a NURBS model for rendering from a software program running on the processor of the computer system. The NURBS model is converted to a Bezier model using the graphics rendering pipeline. A plurality

of Bezier control points are generated from a corresponding plurality of NURBS control points of the NURBS model using a tri-linear interpolator in the graphics pipeline. The plurality of Bezier control points are generated from the corresponding plurality of NURBS control points of the NURBS model by using the plurality of NURBS control points as inputs to the tri-linear interpolator, and evaluating the NURBS control points to obtain each of the plurality of Bezier control points. Additionally, a plurality of points on a curve or surface are generated, wherein the curve or surface is defined by the Bezier model, using the graphics rendering pipeline. The curve or surface defined by the NURBS model is rendered using the plurality of points and using the graphics rendering pipeline such that the curve or surface is rendered without first converting the NURBS model to a polygon mesh.

Appellants respectfully contend that there is no suggestion, motivation, or teaching found in the cited references to combine or modify them and obtain the method of providing a user interface of the present invention as recited in Independent Claim 1. There is no suggestion, motivation, or teaching in the cited references to combine 1) the iterative spline interpolation method for a numerically controlled machine tool device of Jia with 2) the method and apparatus for rendering trimmed NURBS surface representing a mapping from U and V parametric ordinance to X, Y, and Z geometric coordinates of Gharachorloo with 3) the parametric surface evaluation method and apparatus for a computer graphics display system of Luken with 4) the method for generating plane technical curves or contours of Schulmeiss. Additionally, the Examiner relies on the statement that it would have been

obvious to a person of ordinary skill in the art at the time the invention was made to combine the methods described in the cited references, but fails to sufficiently provide specific references to Jia, Gharachorloo, Luken, and Schulmeiss that suggest, motivate, or teach combining or modifying the cited references.

Moreover, the combination of Jia, Gharachorloo, Luken, and Schulmeiss does not teach, suggest , or motivate all the limitations recited in independent Claim 1. Even if the teachings of Jia, Gharachorloo, Luken, and Schulmeiss are combined as described in the Official Actions, the result would be necessarily different from the teachings of Independent Claim 1.

Appellants respectfully assert that the embodiment of the present invention recited in Claim 1 is different from the method and apparatus for rendering trimmed parametric surfaces of Gharachorloo and the method and system for spline interpolation and their use in CNC of Jia. Appellants further assert that even with Luken and Schulmeiss, Gharachorloo and Jia do not show the method of the present invention as recited in Claim 1.

Specifically, the Examiner states that Jia et al teach a method of converting a NURBS surface model to a Bezier surface model, evaluating a plurality of NURBS control points into Bezier control points (Col.3: lines 32-37, Col. 4: lines 38-45), and interpolating a plurality of control points (Col. 5: lines 1-5). In contrast, Appellants respectfully submit that Jia is concerned with the use of spline interpolation for the control of CNC and other numerically

controlled machine tools and is completely different from the rendering method recited in Claim 1. Examiner acknowledges this fact by noting that "Jia et al fail to teach the use of tri-linear interpolator, and a method to receive the data from the server (host processor) and rendering it." Appellants do not understand Jia to disclose or teach the rendering of a NURBS defined surface, as recited in the claimed invention. Appellants respectfully assert that this is due to the fact that Jia is directed towards the control of numerically controlled machines, as opposed to the rendering of curves or surfaces.

Additionally, Appellants do not understand Figure 5 of Jia, or the description thereof, to describe a method for rendering a NURBS defined curve or surface without first converting the NURBS defined curve or surface to a polygon mesh, as asserted in previous Office Actions. Column 8, lines 4-16 of Jia explicitly describe Figure 5 as showing "the same cubic curve shown in FIG. 2, described instead by its equivalent Bezier control polygons. "

The Examiner further states that Gharachorloo et al teaches "a method to receive data from a host processor into a graphics pipeline, and use the graphics pipeline to render the object (refer Figs. 1, 2, 2A)." Appellants respectfully submit that Gharachorloo specifically recites converting the parametric representation into a polygon mesh prior to rendering (Gharachorloo Col. 2 lines 25-57). This is exactly opposite the method of the present invention as recited in Claim 1 wherein the NURBS model is rendered directly by the graphics pipeline without first being converted to a polygon mesh. Therefore, Gharachorloo teaches away from the claimed invention.

The Examiner also states that Luken "inherently teaches the use of tri-linear interpolators by disclosing that the de Casteljau process performs a linear interpolation between the components (x,y,z) of the control points (Col. 4: lines 40-54; Col. 15: lines 32-45), the NURBS control points forming the input to these interpolators." The Examiner further states that Schulmeiss "discloses the use of de Casteljau algorithm to calculate Bezier control points (Col. 2: lines 16-22)" and according to the rejection, it would be obvious to one skilled in the art at the time the invention was made to store the surface model data in a host processor, and use this method to download the data as needed. Appellants respectfully traverse.

Appellants respectfully submit that the use of the graphics rendering pipeline to natively render NURBS models is not "inherent" or obvious.

As recited in Claim 1, the present invention functions by using the graphics pipeline hardware to render NURBS curves or surfaces. The hardware of the graphics pipeline (e.g., texture mapping units, etc.) are used to directly render the NURBS curve or surface without intermediate conversion to a polygon mesh as required by Gharachorloo. The cited prior art references do not natively render NURBS models with the dedicated rendering hardware of the graphics pipeline. For example, in Gharachorloo, the NURBS models are transformed into polygon meshes by software executing on the host processor (e.g., a CPU subsystem 302).

Claim 1, however, explicitly recites utilizing the graphics rendering pipeline to implement a method of natively rendering NURBS models. The hardware of the graphics rendering pipeline (e.g., raster units, texture mapping units, and texture memory, etc.) is designed and optimized for processing conventional graphics primitives and texel data. As recited in Claim 1, the present invention utilizes this existing hardware to natively render NURBS models without first converting them to polygon meshes. The process of the present invention uses tri-linear interpolation to process NURBS or Bezier control points, instead of texels. First, in the case of a Bezier curve, the process of the present invention uses a graphics pipeline's tri-linear interpolator to generate points on the surface defined by the Bezier model. The Bezier control points define a surface. Once the points on the surface are generated, they are fed back as graphics primitives and subsequently rendered into the surface defined by the Bezier model. The process of rendering the points on the curve or surface occurs in the graphics rendering pipeline, without using the CPU subsystem or data transfer bandwidth between main memory and the CPU subsystem or graphics co-processor (as required if the curve or surface is first converted to a polygon mesh). As such, Appellants respectfully submit that the use of the graphics rendering pipeline to natively render NURBS models is not "inherent" or obvious.

With respect to the combination of references (e.g., Jia, Gharachorloo, Luken, and Schulmiess) to obtain the present 35 U.S.C. Section 103 rejections, Applicant respectfully submits that the Section 103 rejections of the above

referenced Office Action are based on impermissible use of hindsight in view of the invention as claimed and the obviousness rejection is not based in prior art. Applicant finds no passage within the cited references which suggest or teach the claimed invention. Applicant respectfully asserts that such rejection is legally improper. Rockwell International Corp. v. United States, 147 F.3d 1358 (Fed. Cir. 1998). It is error to reconstruct the patentee's claimed invention from the prior art using the patentee's claim as a "blueprint." Interconnect Planning Corp. v. Feil, 774 F.2d 1132 (Fed. Cir. 1985).

Accordingly, for the above reasoning, Appellants respectfully submit that the present invention as recited in independent Claim 1 is not rendered unpatentable within the meaning of 35 U.S.C. § 103(a) by the Jia, Gharachorloo, Luken, and Schulmeiss references alone or in combination.

Dependent Claims 6 and 7

Dependent claims 6 includes the limitations of independent Claim 1. As discussed above, independent Claim 1 is not rendered obvious by the cited references and is allowable over the teachings, motivation, or suggestions of Jia, Gharachorloo, Luken, and Schulmeiss. Dependent claims 6 further recites limitations wherein step c) further includes the step of generating a plurality of points on the curve or surface using a plurality of Bezier control points, the Bezier control points generated from a corresponding plurality of NURBS control points using a tri-linear interpolator in the graphics pipeline.
Dependent Claim 7 includes the limitations of dependent claims 6 and further recites limitations for the plurality of Bezier control points being used as inputs

to a tri-linear interpolator, and evaluating the plurality of Bezier control points to obtain the plurality of points on the curve or surface.

Appellants assert that the combination of Jia, Gharachorloo, Luken, and Schulmeiss does not teach, suggest , or motivate all the limitations recited in independent Claim 1. The Examiner acknowledges this fact and cites a fifth reference, Sherman et al., (U.S. Patent No. 5,734,756), and states that Sherman teaches Bezier curves are "generally evaluated using the recursive algorithm due to de Casteljau". Appellants do not understand Sherman to teach the generation of Bezier control points from a corresponding plurality of NURBS control points using a tri-linear interpolator, the plurality of Bezier control points being used as inputs to a tri-linear interpolator, and evaluating the plurality of Bezier control points to obtain the plurality of points on the curve or surface as recited in Claim 7.

Additionally, the Examiner relies on the statement that it would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the methods described in the cited references, but fails to sufficiently provide specific references to Jia, Gharachorloo, Luken, Schulmeiss, and Sherman that suggest, motivate, or teach combining or modifying the cited references to obtain the invention as recited in Claim 7.

Issue 2: Whether Claims 9 and 12 are unpatentable under 35 U.S.C. § 103(a) over Jia et al., U.S. Patent No. 5,726,896 in view of Luken, Jr. U.S. Patent No. 5,278,948 in further view of Schulmeiss, U.S.

Patent No. 5,717,847 and in view of Sherman et al., U.S. Patent No. 5,734,756.

Independent Claim 9 recites a computer implemented method for rendering a NURBS defined curve or surface without first converting the NURBS defined curve or surface to a polygon mesh. As with independent Claim 1, the method of Claim 9 is implemented using a graphics rendering pipeline of a computer system. Claim 9 recites the step of implementing a de Casteljau process in the graphics pipeline. A Bezier curve or surface is evaluated using the de Casteljau process implemented using a tri-linear interpolator included in the graphics pipeline by loading inputs of the tri-linear interpolator with a plurality of control points of the Bezier curve or surface, and generating a plurality of points on the curve or surface using the tri-linear interpolator. The Bezier curve or surface is rendered without first converting the Bezier curve or surface to a polygon mesh.

Appellants respectfully contend that there is no suggestion, motivation, or teaching found in the cited references to combine or modify them and obtain the method of providing a user interface of the present invention as recited in Independent Claim 9. There is no suggestion, motivation, or teaching in the cited references to combine 1) the iterative spline interpolation method for a numerically controlled machine tool device of Jia with 2) the parametric surface evaluation method and apparatus for a computer graphics display system of Luken with 3) the method for generating plane technical curves or contours of Schulmeiss with 4) the method for generating a vector

representation of a skeletal configuration of an elongate graphic image of Sherman. Additionally, the Examiner relies on the statement that it would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the methods described in the cited references, but fails to sufficiently provide specific references to Jia, Luken, and Schulmeiss and Sherman that suggest, motivate, or teach combining or modifying the cited references.

Moreover, the combination of Jia, Luken, Sherman, and Schulmeiss does not teach, suggest , or motivate all the limitations recited in independent Claim 9. Even if the teachings of Jia, Luken, Sherman, and Schulmeiss are combined as described in the Official Actions, the result would be necessarily different from the teachings of Independent Claim 9.

Appellants find no teaching in the cited combination for implementing the de Casteljau process using a tri-linear interpolator included in the graphics pipeline by loading inputs of the tri-linear interpolator with a plurality of control points of the Bezier curve or surface and generating a plurality of points on the curve or surface by using the tri-linear interpolator. The cited section of Schulmeiss (Col. 2: lines 16-22) and Luken Col. 4: lines 40-54; Col. 15: lines 32-45) do not show these limitations or render these limitations inherent as stated by the Examiner.

Appellants respectfully assert that the embodiment of the present invention recited in Claim 9 is different from the method and system for spline

interpolation and their use in CNC of Jia. Appellants further assert that even with Luken and Schulmeiss, Jia does not show the method of the present invention as recited in Claim 9.

Specifically, the Examiner states that Jia et al teach a method of converting a NURBS surface model to a Bezier surface model, evaluating a plurality of NURBS control points into Bezier control points (Col.3: lines 32-37, Col. 4: lines 38-45), and interpolating a plurality of control points (Col. 5: lines 1-5). In contrast, Appellants respectfully submit that Jia is concerned with the use of spline interpolation for the control of CNC and other numerically controlled machine tools and is completely different from the rendering method recited in Claim 1. Examiner acknowledges this fact by noting that "Jia et al fail to teach the use of tri-linear interpolator, and a method to receive the data from the server (host processor) and rendering it." Appellants do not understand Jia to disclose or teach the rendering of a NURBS defined surface, as recited in the claimed invention. Appellants respectfully assert that this is due to the fact that Jia is directed towards the control of numerically controlled machines, as opposed to the rendering of curves or surfaces.

Additionally, Appellants do not understand Figure 5 of Jia, or the description thereof, to describe a method for rendering a NURBS defined curve or surface without first converting the NURBS defined curve or surface to a polygon mesh, as asserted in paragraph 5 of the above referenced Office Action. Column 8, lines 4-16 of Jia explicitly describe Figure 5 as showing "the

same cubic curve shown in FIG. 2, described instead by its equivalent Bezier control polygons. "

The Examiner also states that Luken "inherently teaches the use of tri-linear interpolators by disclosing that the de Casteljau process performs a linear interpolation between the components (x,y,z) of the control points (Col. 4: lines 40-54; Col. 15: lines 32-45), the NURBS control points forming the input to these interpolators." The Examiner further states that Schulmeiss "discloses the use of de Casteljau algorithm to calculate Bezier control points (Col. 2: lines 16-22)" and according to the rejection, it would be obvious to one skilled in the art at the time the invention was made to store the surface model data in a host processor, and use this method to download the data as needed. Appellants respectfully traverse.

Appellants respectfully submit that the use of the graphics rendering pipeline to natively render NURBS models is not "inherent" or obvious. As recited in Claim 9, the present invention functions by using the graphics pipeline hardware to render NURBS curves or surfaces. The hardware of the graphics pipeline (e.g., texture mapping units, etc.) are used to directly render the NURBS curve or surface without intermediate conversion of an intermediate Bezier model to a polygon mesh. The cited Prior art references do not natively render Bezier models with the dedicated rendering hardware of the graphics pipeline without intermediate conversion into polygon meshes.

As recited in Claim 9, the present invention utilizes this existing hardware to natively render Bezier curves or surfaces without first converting them to polygon meshes. The process of the present invention uses tri-linear interpolation to process Bezier control points, instead of texels. First, in the case of a Bezier curve, the process of the present invention uses a graphics pipeline's tri-linear interpolator to generate points on the surface defined by the Bezier model. The Bezier control points define a surface. Once the points on the surface are generated, they are fed back as graphics primitives and subsequently rendered into the surface defined by the Bezier model. The process of rendering the points on the curve or surface occurs in the graphics rendering pipeline, without using the CPU subsystem or data transfer bandwidth between main memory and the CPU subsystem or graphics co-processor (as required if the curve or surface is first converted to a polygon mesh). As such, Appellants respectfully submit that the use of the graphics rendering pipeline to natively render Bezier models is not "inherent" or obvious.

With respect to the combination of references (e.g., Jia, Luken, Sherman, and Schulmiess) to obtain the present 35 U.S.C. Section 103 rejections, Applicant respectfully submits that the Section 103 rejections of the above referenced Office Action are based on impermissible use of hindsight in view of the invention as claimed and the obviousness rejection is not based in prior art. Applicant finds no passage within the cited references which suggest or teach the claimed invention. Applicant respectfully asserts that such rejection is legally improper. Rockwell International Corp. v. United States, 147 F.3d 1358 (Fed. Cir. 1998). It is error to reconstruct the patentee's

claimed invention from the prior art using the patentee's claim as a "blueprint." Interconnect Planning Corp. v. Feil, 774 F.2d 1132 (Fed. Cir. 1985).

Accordingly, for the above reasoning, Appellants respectfully submit that the present invention as recited in independent Claim 9 and dependent Claim 12 is not rendered unpatentable within the meaning of 35 U.S.C. § 103(a) by the Jia, Sherman, Luken, and Schulmeiss references alone or in combination.

Issue 3: Whether Claims 16, 18, and 19 are unpatentable under 35 U.S.C. § 103(a) over Jia et al., U.S. Patent No. 5,488,684 in view of Luken, Jr. U.S. Patent No. 5,278,948 in further view of Schulmeiss, U.S. Patent No. 5,717,847.

Independent Claim 16

Independent Claim 16 recites a computer implemented method for generating normal vectors (normals) for a surface using the graphics rendering pipeline of a computer system. Claim 16 recites the step of generating a plurality of surface partials from the surface by loading inputs of a tri-linear interpolator included in a graphics rendering pipeline with a plurality of Bezier control points defining the surface. A plurality of surface tangents are generated from the plurality of surface partials using the graphics rendering pipeline. At least one normal is generated from the plurality of surface tangents using the graphics rendering pipeline.

Appellants respectfully contend that there is no suggestion, motivation, or teaching found in the cited references to combine or modify them and obtain the method of providing a user interface of the present invention as recited in Independent Claim 16. There is no suggestion, motivation, or teaching in the cited references to combine 1) the iterative spline interpolation method for a numerically controlled machine tool device of Jia with 2) the parametric surface evaluation method and apparatus for a computer graphics display system of Luken with 3) the method for generating plane technical curves or contours of Schulmeiss. Additionally, the Examiner relies on the statement that it would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the methods described in the cited references, but fails to sufficiently provide specific references to Jia, Luken, and Schulmeiss that suggest, motivate, or teach combining or modifying the cited references.

Moreover, the combination of Jia, Luken, and Schulmeiss does not teach, suggest , or motivate all the limitations recited in independent Claim 16. Even if the teachings of Jia, Luken, and Schulmeiss are combined as described in the Official Actions, the result would be necessarily different from the teachings of Independent Claim 16.

Appellants respectfully assert that the embodiment of the present invention recited in Claim 16 is different from the parametric surface evaluation method and apparatus for a computer graphics display system of

Luken. Appellants further assert that even with Jia and Schulmeiss, Luken does not show the method of the present invention as recited in Claim 16.

Specifically, the Examiner states that Luken "inherently teaches the use of tri-linear interpolators by disclosing that the de Casteljau process performs a linear interpolation between the components (x,y,z) of the control points (Col. 4: lines 40-54; Col. 15: lines 32-45), the NURBS control points forming the input to these interpolators." The Examiner further states that Schulmeiss "discloses the use of de Casteljau algorithm to calculate Bezier control points (Col. 2: lines 16-22)" and according to the rejection, it would be obvious to one skilled in the art at the time the invention was made to store the surface model data in a host processor, and use this method to download the data as needed. Appellants respectfully traverse.

Appellants respectfully submit that the use of the graphics rendering pipeline to natively render NURBS models is not "inherent" or obvious.

As recited in Claim 16, the present invention functions by using the graphics pipeline hardware to render Bezier curves or surfaces. The hardware of the graphics pipeline (e.g., texture mapping units, etc.) are used to directly render the Bezier curve or surface. The cited Prior art references do not natively render Bezier models with the dedicated rendering hardware of the graphics pipeline. Appellants do not understand the cited reference to teach generating a plurality of surface partials from the surface by loading inputs of a tri-linear interpolator included in a graphics rendering pipeline with a plurality

of Bezier control points defining the surface. Appellants do not understand the cited reference to teach generating a plurality of surface tangents from the plurality of surface partials using the graphics rendering pipeline. Appellants do not understand the cited reference to teach generating at least one normal from the plurality of surface tangents using the graphics rendering pipeline. The hardware of the graphics rendering pipeline (e.g., raster units, texture mapping units, and texture memory, etc.) is designed and optimized for processing conventional graphics primitives and texel data. As recited in Claim 16, the present invention utilizes this existing hardware to render Bezier models using tri-linear interpolation to process Bezier or Bezier control points, instead of texels. First, in the case of a Bezier curve, the process of the present invention uses a graphics pipeline's tri-linear interpolator to generate points on the surface defined by the Bezier model. The Bezier control points define a surface. Once the points on the surface are generated, they are fed back as graphics primitives and subsequently rendered into the surface defined by the Bezier model. The process of rendering the points on the curve or surface occurs in the graphics rendering pipeline, without using the CPU subsystem or data transfer bandwidth between main memory and the CPU subsystem or graphics co-processor (as required if the curve or surface is first converted to a polygon mesh). As such, Appellants respectfully submit that the use of the graphics rendering pipeline to natively render Bezier models is not "inherent" or obvious.

With respect to the combination of references (e.g., Jia, Luken, and Schulmiess) to obtain the present 35 U.S.C. Section 103 rejections, Applicant

respectfully submits that the Section 103 rejections of the above referenced Office Action are based on impermissible use of hindsight in view of the invention as claimed and the obviousness rejection is not based in prior art. Applicant finds no passage within the cited references which suggest or teach the claimed invention. Applicant respectfully asserts that such rejection is legally improper. Rockwell International Corp. v. United States, 147 F.3d 1358 (Fed. Cir. 1998). It is error to reconstruct the patentee's claimed invention from the prior art using the patentee's claim as a "blueprint." Interconnect Planning Corp. v. Feil, 774 F.2d 1132 (Fed. Cir. 1985).

Accordingly, for the above reasoning, Appellants respectfully submit that the present invention as recited in independent Claim 16 is not rendered unpatentable within the meaning of 35 U.S.C. § 103(a) by the Jia, Luken, and Schulmeiss references alone or in combination.

Dependent Claims 18 and 19

Dependent claims 18 and 19 include the limitations of independent Claim 16. As discussed above, independent Claim 16 is not rendered obvious by the cited references and is allowable over the teachings, motivation, or suggestions of Jia, Luken, and Schulmeiss. Dependent Claim 18 further recites limitations wherein the step of generating the plurality of surface tangents from the plurality of surface partials is performed using a blender included in the graphics rendering pipeline. Dependent Claim 19 includes the limitations of dependent Claim 18 and further recites limitations the step of generating the at least one normal from the plurality of surface tangents using

the blender. Appellants assert that the combination of Jia, Luken, and Schulmeiss does not teach, suggest , or motivate all the limitations recited in dependent Claims 16 and 18 and .

Issue 4: Whether Claims 20-25 are unpatentable under 35 U.S.C. § 103(a) over Luken, Jr. U.S. Patent No. 5,278,948 in view of Gharachorloo et al., U.S. Patent No. 5,488,684.

Independent Claim 20

Independent Claim 20 recites a computer implemented method for using the graphics rendering pipeline to render a curve or surface directly from a NURBS (non-uniform rational B-spline) model. Claim 20 recites the step of performing a global to local transformation on a NURBS model using the graphics rendering pipeline. A plurality of NURBS control points are evaluated using tri-linear interpolation in the graphics rendering pipeline to obtain a plurality of points on a curve or surface defined by the NURBS model. The curve or surface is rendered using the plurality of points.

Appellants respectfully contend that the claimed invention is not rendered obvious within the meaning of 35 U.S.C. Section 103(a) to a person having ordinary skill in the art to which the claimed invention pertains by the cited references.

Appellants respectfully contend that there is no suggestion, motivation, or teaching found in the cited references to combine or modify them and obtain

the method of providing a user interface of the present invention as recited in Independent Claim 20. There is no suggestion, motivation, or teaching in the cited references to combine 1) the parametric surface evaluation method and apparatus for a computer graphics display system of Luken with 2) the method and apparatus for rendering trimmed NURBS surface representing a mapping from U and V parametric ordinance to X, Y, and Z geometric coordinates of Gharachorloo. Additionally, the Examiner relies on the statement that it would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the methods described in the cited references, but fails to sufficiently provide specific references to Gharachorloo and Luken that suggest, motivate, or teach combining or modifying the cited references.

Moreover, the combination of Luken and Gharachorloo does not teach, suggest , or motivate all the limitations recited in independent Claim 20. Even if the teachings of Gharachorloo and Luken are combined as described in the Official Actions, the result would be necessarily different from the teachings of Independent Claim 20.

The Examiner states that Gharachorloo et al teaches "a method to receive data from a host processor into a graphics pipeline, and use the graphics pipeline to render the object (refer Figs. 1, 2, 2A)." Appellants respectfully submit that Gharachorloo specifically recites converting the parametric representation into a polygon mesh prior to rendering (Gharachorloo Col. 2 lines 25-57). This is exactly opposite the method of the

present invention as recited in Claim 20 wherein the NURBS model is rendered directly by the graphics pipeline without first being converted to a polygon mesh. Therefore, Gharachorloo teaches away from the claimed invention.

The Examiner also states that Luken "inherently teaches the use of tri-linear interpolators by disclosing that the de Casteljau process performs a linear interpolation between the components (x,y,z) of the control points (Col. 4: lines 40-54; Col. 15: lines 32-45), the NURBS control points forming the input to these interpolators." Appellants respectfully traverse.

Appellants respectfully submit that the use of the graphics rendering pipeline to directly render NURBS models is not "inherent" or obvious.

As recited in Claim 20, the present invention functions by using the graphics pipeline hardware to render NURBS curves or surfaces. The hardware of the graphics pipeline (e.g., texture mapping units, etc.) are used to directly render the NURBS curve or surface without intermediate conversion to a polygon mesh as required by Gharachorloo. The cited Prior art references do not directly render NURBS models with the dedicated rendering hardware of the graphics pipeline. For example, in Gharachorloo, the NURBS models are transformed into polygon meshes by software executing on the host processor (e.g., a CPU subsystem 302). Claim 20, however, explicitly recites utilizing the graphics rendering pipeline to implement a method of directly rendering NURBS models. The hardware of the graphics rendering pipeline (e.g., raster units, texture mapping units, and texture memory, etc.) is designed and

optimized for processing conventional graphics primitives and texel data. The process of the present invention uses tri-linear interpolation to process NURBS or Bezier control points, instead of texels. The process of rendering the points on the curve or surface occurs in the graphics rendering pipeline, without using the CPU subsystem or data transfer bandwidth between main memory and the CPU subsystem or graphics co-processor (as required if the curve or surface is first converted to a polygon mesh). As such, Appellants respectfully submit that the use of the graphics rendering pipeline to natively render NURBS models is not "inherent" or obvious.

Additionally, Claim 20 recites limitations wherein a global to local transformation is performed on a NURBS model using the graphics rendering pipeline. Applicant finds no passage within the cited references which suggest or teach the inclusion of this limitation into the teaching of Luken.

With respect to the combination of references to obtain the present 35 U.S.C. Section 103 rejections, Applicant respectfully submits that the Section 103 rejections of the above referenced Office Action are based on impermissible use of hindsight in view of the invention as claimed and the obviousness rejection is not based in prior art. Applicant finds no passage within the cited references which suggest or teach the claimed invention. Applicant respectfully asserts that such rejection is legally improper. Rockwell International Corp. v. United States, 147 F.3d 1358 (Fed. Cir. 1998). It is error to reconstruct the patentee's claimed invention from the prior art

using the patentee's claim as a "blueprint." Interconnect Planning Corp. v. Feil, 774 F.2d 1132 (Fed. Cir. 1985).

Accordingly, for the above reasoning, Appellants respectfully submit that the present invention as recited in independent Claim 20 is not rendered unpatentable within the meaning of 35 U.S.C. § 103(a) by the Gharachorloo and Luken references, alone or in combination.

Dependent Claims 23-25

Dependent claims 23-25 include the limitations of independent Claim 20. As discussed above, independent Claim 20 is not rendered obvious by the cited references and is allowable over the teachings, motivation, or suggestions of Gharachorloo and Luken. Dependent Claim 23 further recites limitations including the step of indexing the at least one look up table with the graphics rendering pipeline to obtain a plurality of parameters to configure the tri-linear interpolator. Dependent Claim 24 includes the limitations of dependent claims 23 and further recites limitations including the steps of implementing a quadri-linear interpolator using said tri-linear interpolator and generating the plurality of control points using said quadri-linear interpolator. Dependent Claim 25 includes the limitations of Claim 20 and further includes the steps of using the graphics rendering pipeline to implement a convolution and using the convolution to obtain the plurality of points on the curve or surface. Appellants assert that the combination of Luken and Gharachorloo does not teach, suggest , or motivate all the limitations recited in dependent Claims 23-25.

APPENDIX

CLAIMS ON APPEAL

1. In a computer system having a processor, a bus, and a graphics rendering pipeline for displaying 3D graphics on a display, a computer implemented method for rendering a NURBS defined curve or surface without first converting the NURBS defined curve or surface to a polygon mesh, the method comprising the computer implemented steps of:

a) receiving a NURBS model for rendering from a software program running on the processor of the computer system ;

b) converting the NURBS model to a Bezier model using the graphics rendering pipeline;

c) generating a plurality of Bezier control points from a corresponding plurality of NURBS control points using a tri-linear interpolator in the graphics pipeline by:

c1) using the plurality of NURBS control points as inputs to the tri-linear interpolator; and

c2) evaluating the NURBS control points to obtain each of the plurality of Bezier control points;

d) generating a plurality of points on a curve or surface, wherein the curve or surface is defined by the Bezier model, using the graphics rendering pipeline; and

e) rendering the curve or surface defined by the NURBS model using the plurality of points and using the graphics rendering pipeline such that the curve or surface is rendered without first converting the NURBS model to a polygon mesh.

2. The method of claim 1 wherein step a) further includes the step of receiving the NURBS model in the graphics rendering pipeline via the bus, wherein the NURBS model defines all of a curve or surface, or a portion of the curve or surface, to be rendered.

6. The method of claim 1 wherein step c) further includes the step of generating a plurality of points on the curve or surface using a plurality of Bezier control points.

7. The method of claim 6 further including the steps of:

using the plurality of Bezier control points as inputs to a tri-linear interpolator; and

evaluating the plurality of Bezier control points to obtain the plurality of points on the curve or surface.

8. The method of claim 1 further including the steps of:

processing the plurality of points with the graphics rendering pipeline; and

rendering the curve or surface with the graphics rendering pipeline.

9. In a graphics rendering pipeline of a computer system, a method for rendering NURBS defined curves or surfaces using the graphics rendering pipeline without first converting the NURBS defined curve or surface to a polygon mesh, the method comprising the steps of:

a) implementing a de Casteljau process in the graphics pipeline;

b) evaluating a Bezier curve or surface using the de Casteljau process; and

c) implementing the de Casteljau process using a tri-linear interpolator included in the graphics pipeline by:

loading inputs of the tri-linear interpolator with a plurality of control points of the Bezier curve or surface; and

generating a plurality of points on the curve or surface using the tri-linear interpolator; and

d) rendering the Bezier curve or surface without first converting the Bezier curve or surface to a polygon mesh.

12. The method of claim 9 further including the step of using the plurality of points to render the Bezier curve or surface.

13. In a graphics rendering pipeline of a computer system, a method for converting a NURBS (non-uniform rational B-spline) curve or surface to a Bezier curve or surface using the graphics rendering pipeline, the method comprising the steps of:

a) loading a plurality of NURBS control points of a NURBS curve or surface into the graphics rendering pipeline;

b) evaluating the plurality of control points into a resulting plurality of Bezier control points using a tri-linear interpolator included in the graphics rendering pipeline;

c) generating a Bezier curve or surface using the resulting plurality of Bezier control points; and

d) rendering the Bezier curve or surface using a plurality of vertices derived from the plurality of Bezier control points.

14. The method of claim 13 further including the steps of:

loading the plurality of NURBS control points into inputs of a tri-linear interpolator included in the graphics rendering pipeline; and

evaluating the plurality of NURBS control points into the resulting plurality of Bezier control points using the tri-linear interpolator.

15. The method of claim 13 further including the step of transforming the NURBS curve or surface from a global domain to a local domain.

16. In a graphics rendering pipeline of a computer system, a method for generating normal vectors (normals) for a surface, the method comprising the steps of:

a) generating a plurality of surface partials from the surface by loading inputs of a tri-linear interpolator included in a graphics rendering pipeline with a plurality of Bezier control points defining the surface;

b) generating a plurality of surface tangents from the plurality of surface partials using the graphics rendering pipeline; and

c) generating at least one normal from the plurality of surface tangents using the graphics rendering pipeline.

18. The method of claim 16 further including the step of generating the plurality of surface tangents from the plurality of surface partials using a blender included in the graphics rendering pipeline.

19. The method of claim 18 further including the step of generating the at least one normal from the plurality of surface tangents using the blender.

20. In a graphics rendering pipeline of a computer system, a method of using the graphics rendering pipeline to render a curve or surface directly from a NURBS (non-uniform rational B-spline) model, the method comprising the steps of:

a) performing a global to local transformation on a NURBS model using the graphics rendering pipeline;

b) evaluating a plurality of NURBS control points using tri-linear interpolation in the graphics rendering pipeline to obtain a plurality of points on a curve or surface defined by the NURBS model; and

c) rendering the curve or surface using the plurality of points.

21. The method of claim 20 further including the step of indexing at least one look up table within the graphics rendering pipeline to perform the global to local transformation of the NURBS model.

22. The method of claim 21 further including the step of evaluating the plurality of NURBS control points using a tri-linear interpolator included in the graphics rendering pipeline.

23. The method of claim 22 further including the step of indexing the at least one look up table with the graphics rendering pipeline to obtain a plurality of parameters to configure the tri-linear interpolator;

24. The method of claim 23 further including the steps of:

implementing a quadri-linear interpolator using said tri-linear interpolator; and

generating the plurality of control points using said quadri-linear interpolator.

25. The method of claim 20 wherein step b) further includes the steps of:

using the graphics rendering pipeline to implement a convolution; and

using the convolution to obtain the plurality of points on the curve or surface.